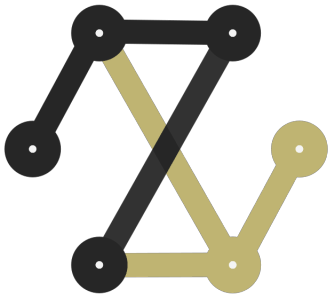

LibreIM

Introducción a PAC Learning

Ignacio Cordón Castillo

14 de enero de 2017

Índice

Estos apuntes son una adptación en su mayoría del contenido del libro [1]

1. Introducción

Damos unas notaciones/definiciones básicas que utilizaremos de aquí en adelante.

- **Dominio**: $\mathcal{X}$, sobre el que tenemos definida una σ álgebra de conjuntos. Llamamos una instancia a $x \in \mathcal{X}$
- **Conjunto de etiquetas**: $\mathcal{Y}$ consideramos $\{0, 1\}$, lo que nos restringe al paradigma binario.
- **Verdadero etiquetado**: Asumimos la existencia de una función $f : \mathcal{X} \to \mathcal{Y}$ que devuelve el verdadero etiquetado de todas las instancias.
- **Generación de instancias**: Asumimos la existencia de una distribución de probabilidad $\mathcal{D}$ sobre $\mathcal{X}$, para la σ álgebra de conjuntos mencionada anteriormente, que nos da información sobre la probabilidad de extraer cada posible instancia desde $\mathcal{X}$.
- **Conjunto/Secuencia de entrenamiento**: $S = ((x_1, y_1), \dots (x_m, y_m))$ secuencia con cada elemento perteneciente a $\mathcal{X} \times \mathcal{Y}$. A veces lo llamaremos conjunto, por abuso de notación, pero se trata de una tupla en $(\mathcal{X} \times \mathcal{Y})^m$ en la que pueden repetirse ejemplos. Podemos ver el conjunto de entrenamiento como una m.a.s (muestra aleatoria simple) $(\mathcal{X}_1, \dots \mathcal{X}_m)$, idéntica e independientemente distribuida, donde cada X_i sigue la misma distribución que $\mathcal{X}$, $X_i \sim \mathcal{D}$ y se etiqueta por f. Lo notaremos $S \sim \mathcal{D}^m$, por abuso de notación.
- **Resultado del aprendizaje**: una función $h : \mathcal{X} \to \mathcal{Y}$ que llamaremos hipótesis/clasificador. Se usa la notación $A(S)$ para denotar la hipótesis que un algoritmo A devuelve para una secuencia de entrenamiento S.
- **Error del clasificador**: Definimos el error del clasificador, suponiendo $\{x \in \mathcal{X} : h(x) \neq f(x)\}$ en la σ álgebra, como:

$$L_{D,f}(h) := P_{x \sim \mathcal{D}}[h(x) \neq f(x)]$$

Por simplificar la escritura, omitiremos a partir de ahora el hecho de que sobre $\mathcal{X}$ tenemos una σ álgebra de conjuntos, y que todas las distribuciones

asignan probabilidad convenientemente a los conjuntos de dicha σ álgebra. Además, consideraremos que la función de verdadero etiquetado y los clasificadores son funciones medibles para que la definición de los errores sean correctas.

1.1. Minimización del riesgo empírico (ERM)

Definición 1. ***Riesgo empírico (ER)***

Definimos el riesgo empírico o error empírico como:

$$L_S(h) = \frac{\#\{i \in 1 \dots m : h(x_i) \neq y_i\}}{m}$$

Podemos pensar en él como el error del clasificador sobre el conjunto de entrenamiento. Un algoritmo que obtiene hipótesis que minimizan el error empírico recibe el nombre de ERM y notamos $ERM(S)$ al clasificador que obtiene, para un determinado conjunto de entrenamiento $S \sim \mathcal{D}^m$.

Este error no es siempre óptimo. Pensemos en el siguiente ejemplo:

Sea $\mathcal{X} = \mathbb{R}$, $\mathcal{D}$ la distribución uniforme sobre $[0, 2] \subset \mathbb{R}$, y la siguiente función:

$$f(x) = \begin{cases} 1 & x \in [0, 1] \\ 0 & x \in \mathbb{R} \setminus [0, 1] \end{cases}$$

$S = ((x_1, y_1), \dots (x_m, y_m))$ un conjunto de entrenamiento de tamaño m y el clasificador:

$$h_S(x) = \begin{cases} y_i & \exists i \in \{1 \dots m\} : x = x_i \\ 0 & \nexists i \in \{1 \dots m\} : x = x_i \end{cases}$$

Nótese que el conjunto de entrenamiento no puede tener elementos no repetidos puesto que se etiquetan mediante f, que es una función y no puede arrojar dos imágenes distintas para un mismo $x \in \mathcal{X}$ de entrada.

Este clasificador es perfecto respecto a la minimización de riesgo empírico, pero $L_{\mathcal{D},f}(h_S) = 1/2$. Es decir, tiene el mismo nivel de acierto que el clasificador idénticamente 1. A este fenómeno, minimizar el riesgo empírico siendo un clasificador con un error muy alto, lo denominamos **overfitting**.

El hecho de tomar el error sobre el conjunto de entrenamiento como aproximación al verdadero error del clasificador se respalda por la siguiente proposición:

Proposición 1. ***Relación entre ER y error del clasificador***

Sea $\mathcal{H}$ clase de clasificadores binarios sobre un dominio $\mathcal{X}$. Sea $\mathcal{D}$ una distribución desconocida sobre $\mathcal{X}$. Sea f una hipótesis objetivo en $\mathcal{H}$. Se fija $h \in \mathcal{H}$. Probar que:

$$\mathbb{E}_{S\sim\mathcal{D}}[L_S(h)] = L_{\mathbb{D},f}(h)$$

Llamamos $P = \mathbb{P}_{x\sim\mathcal{D}}(f(x) \neq h(x))$

$$\begin{aligned}\mathbb{E}_{S\sim\mathcal{D}}[L_S(h)] &= \sum_{k=0}^{m} \frac{k}{m}\binom{m}{k}P^k(1-P)^{m-k} = \sum_{k=1}^{m} \frac{k}{m}\binom{m}{k}P^k(1-P)^{m-k} = \\ &= \sum_{k=1}^{m}\binom{m-1}{k-1}P^k(1-P)^{m-k} = \sum_{k=0}^{m-1}\binom{m-1}{k}P^{k+1}(1-P)^{m-1-k} = \\ &= P\cdot\sum_{k=0}^{m-1}\binom{m-1}{k}P^k(1-P)^{m-1-k} = P(1+(1-P))^{m-1} = P\end{aligned}$$

1.2. ERM con *sesgo inductivo*

Se intenta corregir el ERM restringiendo el espacio de búsqueda, esto es, la clase de hipótesis $\mathcal{H}$ desde la que el algoritmo puede escoger un $h : \mathcal{X} \to \mathcal{Y}$. Llamamos a esto *sesgo inductivo* puesto que se asumirá una determinada clase de funciones $\mathcal{H}$ en función de las características del problema.

Notaremos a un clasificador obtenido con este paradigma $h_S = ERM_{\mathcal{H}}(S)$, y lo definimos de manera que:

$$h_S \in argmin_{h \in \mathcal{H}} L_S(h)$$

Definimos la propiedad de factibilidad, que usaremos más adelante.

Definición 2. ***Propiedad de factibilidad***

Existe $\bar{h} \in \mathcal{H}$ *verificando* $L_{D,f}(\bar{h}) = 0$.

La hipótesis de factibilidad implica que $P_{S \sim \mathcal{D}^m}[L_S(\bar{h}) = 0] = 1$, y por tanto $P_{S \sim \mathcal{D}^m}[L_S(h_S) = 0] = 1$.

El valor $L_{\mathcal{D},f}(h_S)$ dependerá del conjunto de entrenamiento S, y la elección del mismo está sometida al azar. Además, necesitamos una medida de la bondad de una predicción.

2. Aprendizaje PAC.

Definición 3. ***Aprendizaje PAC (Probablemente Aproximadamente Correcto)***

Una clase de funciones definidas sobre $\mathcal{X}$, $\mathcal{H}$ es PAC learnable si existe una función $m_{\mathcal{H}} :]0,1[^2 \to \mathbb{N}$, llamada complejidad muestral, y un algoritmo A verificando que si $0 < \epsilon, \delta < 1$, entonces para toda distribución $\mathcal{D}$ sobre $\mathcal{X}$ y para toda función de verdadero etiquetado $f : \mathcal{X} \to \{0,1\}$, si la propiedad de factibilidad se cumple, ejecutando el algoritmo para un conjunto de entrenamiento $S \sim \mathcal{D}^m$ etiquetado mediante f, con $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ el algoritmo devuelve una hipótesis $A(S) = h \in \mathcal{H}$ verificando que:

$$P_{S \sim \mathcal{D}^m}[L_{\mathcal{D},f}(h) \leq \epsilon] \geq 1 - \delta$$

$(1-\delta)$ es la *confianza de la predicción* (probablemente) y $(1-\epsilon)$ la *exactitud* (correcto).

Podemos considerar $m_{\mathcal{H}}$ única en el sentido de que para cada (δ, ϵ) nos devuelve el menor natural verificando las hipótesis del enunciado.

Nótese que las condiciones exigidas: cumplir la propiedad de factibilidad y que la hipótesis devuelta deba estar en $\mathcal{H}$ son muy fuertes.

2.1. Aprendizaje con clases finitas

Teorema 1. ***Las clases finitas de funciones son PAC learnable***

Sea $\mathcal{H}$ una clase finita de funciones definidas sobre un conjunto $\mathcal{X}$. Sean $0 < \epsilon, \delta < 1$, y un natural $m \in \mathbb{N}$ verificando:

$$m \geq \frac{1}{\epsilon} log\left(\frac{|\mathcal{H}|}{\delta}\right)$$

Entonces para toda función de verdadero etiquetado $f : \mathcal{X} \to \{0,1\}$, y para toda distribución $\mathcal{X} \sim \mathcal{D}$ para la que se verifique la ***propiedad de factibi-***

lidad *entonces las hipótesis que obtenemos a través del algoritmo ERM son con una confianza superior a* $1-\delta$, $1-\epsilon$ *exactas.*

Como consecuencia, deducimos que la complejidad muestral es menor o igual a $\left\lceil \frac{1}{\epsilon} log\left(\frac{|\mathcal{H}|}{\delta}\right)\right\rceil$

Demostración. Fijada una distribución $\mathcal{D}$ y una función de etiquetado f, notamos:

$$\mathcal{H}_B = \{h \in \mathcal{H} : L_{\mathcal{D},f}(h) > \epsilon\}$$

Se tiene:

$$P_{S\sim\mathcal{D}^m}[L_{\mathcal{D},f}(h_S) > \epsilon] \leq P_{S\sim\mathcal{D}^m}[\exists h \in \mathcal{H}_B : L_S(h) = 0] \leq \sum_{h\in\mathcal{H}_B} P_{S\sim\mathcal{D}^m}[L_S(h) = 0]$$

La primera desigualdad viene dada porque dada h_S se verifica, por la propiedad de factibilidad, que $L_S(h_S) = 0$. La segunda por subaditividad.

Además, fijada $h \in \mathcal{H}_B$, como $L_{\mathcal{D},f}(h) > \epsilon$:

$$P_{S\sim\mathcal{D}^m}[L_S(h) = 0] = P_{(x_1,\dots x_n)\sim\mathcal{D}^m}[\forall i \quad h(x_i) = f(x_i)] =$$
$$= \prod_{i=1}^{m} P_{x\sim\mathcal{D}}[h(x) = f(x)] = \prod_{i=1}^{m}(1 - L_{\mathcal{D},f}(h)) \leq (1-\epsilon)^m \leq e^{-\epsilon m}$$

Las dos desigualdades probadas, junto a la hipótesis del enunciado, y usando $\mathcal{H}_B \subseteq \mathcal{H}$ dan lugar a:

$$P_{S\sim\mathcal{D}^m}[L_{\mathcal{D},f}(h_S) > \epsilon] \leq |\mathcal{H}|e^{-\epsilon m} \leq \delta$$

□

2.2. Aprendizaje con clases no finitas

¿Hay ejemplos de clases infinitas PAC learnable? Veamos un ejemplo.

Definición 4. ***Clasificadores de rectángulo***

Un clasificador de rectángulo es un clasificador que asigna 1 a los puntos que se quedan dentro de un cierto rectángulo en el plano real.

$$h_{a,b,c,d}(x,y) = \begin{cases} 1 & a \leq x \leq b, c \leq y \leq d \\ 0 & si \quad no \end{cases}$$

La clase de clasificadores de rectángulo en el plano se define por:

$$\mathcal{H}^2_{rec} = \{h_{a,b,c,d} : a \leq b, c \leq d\}$$

Proposición 2. ***Los rectángulos son PAC learnables***

Asumiendo propiedad de factibilidad, los rectángulos son PAC learnables

Sea A el algoritmo que devuelve el rectángulo más pequeño que engloba a todos los ejemplos positivos del conjunto de entrenamiento S.

Partiendo de la propiedad de factibilidad, debe existir un clasificador de rectángulo $\bar{h} = h_{a,b,c,d}$ que haga el ERM nulo y que cumpla $L_{\mathcal{D},f}(\bar{h})$. Por tanto debe verificarse que h_S debe contener a todos los ejemplos positivos del conjunto de entrenamiento, ya que si valiese 0 en algún ejemplo positivo del conjunto de entrenamiento, el ERM sería mayor que 0.

El algoritmo que devuelve el mínimo rectángulo que engloba a todos los ejemplos positivos es por tanto un ERM.

Veamos que con este algoritmo minimizador del ERM la clase de rectángulos es PAC learnable.

Sea $R^* = R(a,b,c,d)$ el rectángulo del apartado 1. Entonces $P_{S\sim\mathcal{D}^2}[f(R^*) = \{1\}] = 1$

Tomamos $R_1 = R(a,a^*,c,d)$ un rectángulo que concentra una masa de probabilidad menor o igual a $\epsilon/4$, con $a \leq a^*$.

$R_2 = (b^*, b, c, d), R_3 = (a, b, c, c^*), R_4 = (a, b, d^*, d)$ se definen de forma análoga.

Llamando $h_R = A(S)$, R el rectángulo obtenido como resultado de aplicar el algoritmo del ejercicio. Es claro que con probabilidad 1, $R \subset R^*$.

Si se tiene $\forall i : R \cap R_i \neq \emptyset$:

$$L_{\mathcal{D},f}(h_R) = P_{x\sim\mathcal{D}}[h_R(x) \neq f(x)] \leq P_{x\sim\mathcal{D}}\left(\cup_i [h_R(x) \neq f(x)] \cap R_i\right) \leq$$
$$\leq P_{x\sim\mathcal{D}}\left(\cup_i R_i\right) \leq 4\frac{\epsilon}{4} = \epsilon$$

La demostración acaba probando que:

$$P(\exists i : S \cap R_i = \emptyset) \leq \sum_{i=1}^{4} P(S \cap R_i = \emptyset) = 4(1 - \frac{\epsilon}{4})^m \leq 4e^{-m}$$

3. Generalización aprendizaje PAC: PAC agnóstico

Hasta ahora tenemos dos problemas en la definición de PAC. Intentamos buscar una hipótesis sobre una función de verdadero etiquetado, f determinista, que por tanto no podrá asignar dos imágenes distintas al mismo punto, y además, estamos suponiendo que se cumple la propiedad de factibilidad.

Para paliar esto, podríamos considerar $\mathcal{D}$ como la distribución conjunta sobre $\mathcal{X} \times \mathcal{Y}$, y la noción de error para $h : \mathcal{X} \to \mathcal{Y}$ quedaría:

$$L_{\mathcal{D}}(h) := P_{(x,y)\sim\mathcal{D}}[h(x) \neq y]$$

Con estos conceptos revisitados, podríamos asegurar que la hipótesis que menor error comete para $\mathcal{Y} = \{0, 1\}$ es el llamado **clasificador de Bayes**:

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & P[y = 1|x] >= 0{,}5 \\ 0 & si \quad no \end{cases}$$

Pero deseamos ir aún más allá, y generalizar la definición para una función de pérdida arbitraria.

Definición 5. ***Función de pérdida***

Dados un conjunto $\mathcal{H}$, Z y una σ álgebra de conjuntos sobre Z, se denomina función de pérdida de $\mathcal{H}$ sobre Z a cualquier función de la forma:

$$l : \mathcal{H} \times Z \to \mathbb{R}^+$$

que verifique que la función $l(h, \cdot)$ sea medible $\forall h \in \mathcal{H}$ sobre la σ álgebra inicial.

Aumiendo ya como $\mathcal{D}$ la distribución conjunta, con funciones de pérdida arbitrarias, redefiniríamos los conceptos de *error* y *error empírico* de la forma:

$$L_{\mathcal{D}}(h) := \mathbb{E}_{z\sim\mathcal{D}}[l(h,z)]$$
$$L_S(h) := \frac{1}{m}\sum_{i=1}^{m} l(h,z_i)$$

Definición 6. ***Aprendizaje PAC agnóstico***

Una clase de funciones $\mathcal{H}$ definidas en $\mathcal{X}$ y con imagen en $\mathcal{Y}$ es agnósticamente PAC learnable respecto a $Z = \mathcal{X}\times\mathcal{Y}$ (sobre el que tenemos definida una σ álgebra de conjuntos) y a una función de pérdida $l : \mathcal{H}\times Z \to \mathbb{R}^+$ si existe una función $m_{\mathcal{H}} :]0,1[^2 \to \mathbb{N}$ y un algoritmo A verificando que si $0 < \epsilon, \delta < 1$, entonces para toda distribución $\mathcal{D}$ sobre Z ejecutando el algoritmo para un conjunto de entrenamiento $S \sim \mathcal{D}^m$, con $m \geq m_{\mathcal{H}}(\epsilon,\delta)$ el algoritmo devuelve una hipótesis $A(S) = h \in \mathcal{H}$ verificando que:

$$P_{S\sim\mathcal{D}^m}[L_{\mathcal{D}}(h) \leq inf_{h'\in\mathcal{H}} L_{\mathcal{D}}(h') + \epsilon] \geq 1 - \delta$$

Notamos que esta definición, en caso de cumplirse la propiedad de factibilidad, tomando una **función de pérdida 0-1**:

$$l_{0-1}(h,(x,y)) := \begin{cases} 0 & h(x) = y \\ 1 & si \quad no \end{cases}$$

equivale a la primera definición que dimos de aprendizaje PAC si asumimos propiedad de factibilidad. Por ello no distinguiremos en el uso de uno u otro concepto, sino que se deducirá de si estamos asumiendo propiedad de factibilidad o no.

Cuando permitimos que el algoritmo A devuelva una función $h \notin \mathcal{H}$, de manera que $h \in \mathcal{H}'$ y $\mathcal{H} \subset \mathcal{H}'$ una clase de funciones a donde la función de pérdida es extendible de manera natural, el aprendizaje recibe el nombre de **aprendizaje impropio**. La definición aquí dada se ha hecho para **aprendizaje propio**.

4. Condiciones suficientes para ser PAC learnable

Definición 7. ***Conjunto de entrenamiento ϵ representativo***

Un conjunto de entrenamiento S se dice ϵ representativo respecto a un dominio Z, a una clase de hipótesis $\mathcal{H}$, una función de pérdida l y una distribución $\mathcal{D}$ sobre Z si:

$$\forall h \in \mathcal{H}, |L_S(h) - L_{\mathcal{D}}(h)| \leq \epsilon$$

Lema 1. *Sea un conjunto de entrenamiento de tamaño S, $\frac{\epsilon}{2}$ representativo respecto a un dominio Z, a una clase de hipótesis $\mathcal{H}$, una función de pérdida l y una distribución $\mathcal{D}$. Entonces:*

$$L_{\mathcal{D}}(ERM_{\mathcal{H}}(S)) \leq inf_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$$

Demostración. Para $h \in \mathcal{H}$ arbitrario:

$$L_{\mathcal{D}}(h_S) \leq L_{\mathcal{S}}(h_S) + \frac{\epsilon}{2} \leq L_{\mathcal{S}}(h) + \frac{\epsilon}{2} \leq L_{\mathcal{D}}(h) + \frac{\epsilon}{2} + \frac{\epsilon}{2} = L_{\mathcal{D}}(h) + \epsilon$$

□

Definición 8. ***Convergencia uniforme***

Decimos que una clase de hipótesis $\mathcal{H}$ tiene la propiedad de convergencia uniforme respecto a un dominio Z, y a una función l si para todo $0 < \delta, \epsilon < 1$ existe $m_{\epsilon,\delta}$ verificando que para toda distribución $\mathcal{D}$ sobre Z, si S es un conjunto de entrenamiento de tamaño mayor o igual a $m_{\epsilon,\delta}$, entonces:

$$P_{S \sim \mathcal{D}^m}[\forall h \in \mathcal{H}, |L_S(h) - L_{\mathcal{D}}(h)| \leq \epsilon] \geq 1 - \delta$$

Teorema 2. ***La convergencia uniforme es condición suficiente para ser PAC learnable***

Sea $\mathcal{H}$ una clase de hipótesis con la propiedad de convergencia uniforme. Entonces es PAC learnable con complejidad muestral menor o igual al $m_{\frac{\epsilon}{2},\delta}$ dado en la definición anterior y el algoritmo ERM

Demostración. La prueba es trivial desde el lema y la definición de convergencia uniforme. □

Proposición 3. ***Las clases finitas tienen la propiedad de convergencia uniforme***

Sea $\mathcal{H}$ una clase de hipótesis finita, Z un dominio y sea $l : \mathcal{H} \times Z \to [a, b]$ una función de pérdida. Entonces $\mathcal{H}$ verifica la propiedad de convegencia uniforme con:

$$m_{\epsilon,\delta} \leq \left\lceil \frac{log(2|\mathcal{H}|/\delta)(b-a)^2}{2\epsilon^2} \right\rceil$$

Lema 2. ***Desigualdad de Hoeffding***

Sean $X_1, \dots X_n$ una muestra aleatoria simple de una variable X, $\bar{X} = \frac{1}{m}\sum_{i=1}^{m} X_i$ con $E[\bar{X}] = \mu$ y $P[a \leq X_i \leq b] = 1$. Entonces para todo $\epsilon > 0$

$$P\left[\left|\bar{X} - \mu\right| > \epsilon\right] \leq 2e^{-2m\left(\frac{\epsilon}{(b-a)}\right)^2}$$

Demostración. Sea $\mathcal{H}$ una clase de hipótesis finita.

Fijamos $0 < \delta, \epsilon < 1$. Necesitamos encontrar $m \in \mathbb{N}$ verificando:

$$P_{S\sim\mathcal{D}^m}[\exists h \in \mathcal{H} | L_S(h) - L_{\mathcal{D}}(h)| > \epsilon] < \delta$$

Partimos de la siguiente desigualdad, que usaremos más adelante, obtenida por subaditividad:

$$P_{S\sim\mathcal{D}^m}[\exists h \in \mathcal{H} | L_S(h) - L_{\mathcal{D}}(h)| > \epsilon] \leq \sum_{h\in\mathcal{H}} P_{S\sim\mathcal{D}^m}[|L_S(h) - L_{\mathcal{D}}(h)| > \epsilon]$$

Fijamos $h \in \mathcal{H}$.

Dado un conjunto de entrenamiento $S = (z_1, \dots z_m)$, recordamos que $L_{\mathcal{D}}(h) = \mathbb{E}_{z\sim\mathcal{D}}[l(h, z)]$ y que $L_S(h) = \frac{1}{m}\sum_{i=1}^{m} l(h, z_i)$

Donde $z_i \sim \mathcal{D}$ y por tanto $\mathbb{E}_{S\sim\mathcal{D}^m}[L_S(h)] = \mathbb{E}_{z\sim\mathcal{D}}[l(h,z)] = L_{\mathcal{D}}(h)$. Además, llamando $X_i = l(h, z_i)$, por ser z_i realizaciones muestrales de una m.a.s se tiene que las X_i son independientes e idénticamente distribuidas, con $P[a < X_i < b] = 1$. Estamos en condiciones de aplicar la desigualdad de Hoeffding.

Por tanto:

$$P_{S\sim\mathcal{D}^m}\left[\left|\frac{1}{m}\sum_{i=1}^{m} X_i - L_{\mathcal{D}}(h)\right| > \epsilon\right] = P_{S\sim\mathcal{D}^m}[|L_S(h) - L_{\mathcal{D}}(h)| > \epsilon] \leq 2e^{-2m\left(\frac{\epsilon}{b-a}\right)^2}$$

Y por tanto:

$$P_{S\sim\mathcal{D}^m}[\exists h \in \mathcal{H} | L_S(h) - L_{\mathcal{D}}(h)| > \epsilon] < |\mathcal{H}| 2e^{-2m\left(\frac{\epsilon}{b-a}\right)^2}$$

□

Recordemos hasta ahora el resultado que habíamos obtenido era su carácter PAC learnable, donde agnósticamente PAC learnable y learnable con funciones de pérdida 0-1 era un término equivalente. El teorema que enunciamos a continuación, deducible a partir del teorema sobre el caracter agnóstico - PAC learnable de clases de funciones con propiedad de convergencia uniforme, en particular las finitas, generaliza el resultado para cualquier funciones de pérdida acotada.

Teorema 3. ***Las clases finitas son agnósticamente PAC learnable***

Sea $\mathcal{H}$ una clase de hipótesis finita, Z un dominio y sea $l : \mathcal{H} \times Z \to [a, b]$ una función de pérdida. Entonces $\mathcal{H}$ es PAC learnable con complejidad muestral:

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \left\lceil \frac{2log(2|\mathcal{H}|/\delta)(b-a)^2}{\epsilon^2} \right\rceil$$

5. Equilibrio error-varianza *bias-complexity tradeoff*

Veamos que dado un algoritmo de aprendizaje no puede ser el óptimo para aprender todas las distribuciones.

Damos un lema previo, la desigualdad de Markov:

Lema 3. ***Desigualdad de Markov***

Dada una variable aleatoria Z no negativa. Entonces para todo $a \geq 0$

$$P[Z \geq a] \leq \frac{\mathbb{E}[Z]}{a}$$

Teorema 4. ***Teorema de No Free Lunch***

Sea A cualquier algoritmo de aprendizaje para clasificación binaria con respecto a la función de pérdida 0-1 sobre el dominio $\mathcal{X}$. Sea un conjunto de entrenamiento de tamaño $m < |\mathcal{X}|/2$. Entonces existe una distribución $\mathcal{D}$ sobre $\mathcal{X} \times \{0,1\}$ verificando:

1. *Existe una función $f : \mathcal{X} \to \{0,1\}$ con $L_{\mathcal{D}}(f) = 0$*
2. *$P_{S\sim\mathcal{D}^m}[L_{\mathcal{D}}(A(S)) \geq 1/8] \geq 1/7$*

Demostración. Sea un conjunto de entrenamiento (consideramos un conjunto y no una secuencia) de tamaño $2m$, C. Hay $T = 2^{2m}$ posibilidades de etiquetado del conjunto, esto es, 2^{2m} posibles hipótesis, $f_i : C \to \{0,1\}$, que vamos a extender a $\mathcal{X}$ llamándolas $\bar{f}_i$ de forma que $\bar{f}_{i|C} = f_i$ y $\bar{f}_i(x) = 0 \quad \forall x \in \mathcal{X} \setminus C$. Vamos a tomar para cada una de ellas una distribución $\mathcal{D}_i$ definida sobre $\mathcal{X} \times \{0,1\}$ definida por:

$$\forall (x,y) \in \mathcal{X} \times \{0,1\} \qquad P_{z\sim\mathcal{D}_i}[z = (x,y)] = \begin{cases} 1/|C| & \exists x_i \in C : y = f(x_i) \\ 0 & si \quad no \end{cases}$$

Claramente $L_{\mathcal{D}_i}(f_i) = 0$

Vamos a probar que:

$$\exists i \in \{1, \dots 2m\} : \mathbb{E}_{S \sim \mathcal{D}_i^m}[L_{\mathcal{D}_i}(A(S))] \geq \frac{1}{4}$$

Hay $k = (2m)^m$ posibles secuencias de entrenamiento de tamaño m, $S_j, j = 1, \dots k$ tomadas desde C. Siendo $S_j = (x_1, \dots x_m)$ notamos $S_j^i = ((x_1, f_i(x_1)), \dots, (x_m, f_i(x_m)))$. Cada S_j tiene la misma probabilidad de ser nuestro conjunto de entrenamiento (extracción de m valores con reemplazamiento desde el conjunto C), verificándose:

$$\mathbb{E}_{S \sim \mathcal{D}_i^m}[L_{\mathcal{D}_i}(A(S))] = \frac{1}{k} \sum_{j=1}^{k} L_{\mathcal{D}_i}(A(S_j^i))$$

Recordando que hemos llamado $k = (2m)^m$, $T = 2^{2m}$, se tiene:

$$\begin{aligned} max_{i \in \{1, \dots T\}} \frac{1}{k} \sum_{j=1}^{k} L_{\mathcal{D}_i}(A(S_j^i)) &\geq \frac{1}{T} \sum_{i=1}^{T} \frac{1}{k} \sum_{j=1}^{k} L_{\mathcal{D}_i}(A(S_j^i)) = \\ &= \frac{1}{k} \sum_{j=1}^{k} \frac{1}{T} \sum_{i=1}^{T} L_{\mathcal{D}_i}(A(S_j^i)) \geq \\ &\geq min_{j \in \{1, \dots k\}} \frac{1}{T} \sum_{i=1}^{T} L_{\mathcal{D}_i}(A(S_j^i)) \end{aligned}$$

Además fijado $j \in \{1, \dots k\}$, se tiene que que para todo $i \in \{1, \dots T\}$:

$$L_{\mathcal{D}_i}(h) = \frac{1}{|C|} \sum_{x \in C} \mathbb{1}_{[A(S_j^i)(x) \neq f_i(x)]} = \frac{1}{2m} \sum_{x \in C} \mathbb{1}_{[A(S_j^i)(x) \neq f_i(x)]}$$

Por tanto:

$$\frac{1}{T}\sum_{i=1}^{T} L_{\mathcal{D}_i}(A(S_j^i)) \geq \frac{1}{T}\sum_{i=1}^{T}\frac{1}{2m}\sum_{x\in C}\mathbb{1}_{[A(S_j^i)(x)\neq f_i(x)]} =$$

$$= \frac{1}{2m}\sum_{x\in C}\frac{1}{T}\sum_{i=1}^{T}\mathbb{1}_{[A(S_j^i)(x)\neq f_i(x)]} \geq$$

$$\geq \frac{1}{2} min_{x\in C}\frac{1}{T}\sum_{i=1}^{T}\mathbb{1}_{[A(S_j^i)(x)\neq f_i(x)]}$$

Como dado un $x \in C$ cualquiera, la mitad de clasificadores f_i clasificarán x bien y la otra mitad mal, se tiene:

$$\frac{1}{2} min_{x\in C}\frac{1}{T}\sum_{i=1}^{T}\mathbb{1}_{[A(S_j^i)(x)\neq f_i(x)]} = \frac{1}{2}\frac{1}{T}\frac{T}{2} = \frac{1}{4}$$

Y uniendo toda esta información:

$$max_{i\in\{1,\dots T\}}\frac{1}{k}\sum_{j=1}^{k} L_{\mathcal{D}_i}(A(S_j^i)) \geq \frac{1}{4}$$

Sea $k = argmax_{i\in\{1,\dots T\}}\frac{1}{k}\sum_{j=1}^{k} L_{\mathcal{D}_i}(A(S_j^i))$

Si $\mathcal{D} = \mathcal{D}_k$ cumple la parte 2 del enunciado del teorema, es nuestra distribución buscada, y como función buscada en el apartado 1. podemos tomar $f = f_k$

Como $L_{\mathcal{D}}(A(\cdot))$ puede ser vista como una variable aleatoria donde $S \sim \mathcal{D}^m$ y que toma valores en $[0, 1]$, tenemos que tomando $Z = 1 - L_{\mathcal{D}}(A(\cdot))$, $a = \frac{7}{8}$ en el lema previo llegamos a:

$$P_{S\sim\mathcal{D}^m}\left(\frac{1}{8} \geq L_{\mathcal{D}}(A(S))\right) \leq \frac{3}{4}\cdot\frac{8}{7} = 24/28$$

donde $\mathbb{E}(Z) = \mathbb{E}(1 - L_{\mathcal{D}}(A(\cdot))) = 1 - \mathbb{E}(L_{\mathcal{D}}(A(\cdot))) \leq \frac{3}{4}$

Es decir:

$$P_{S\sim\mathcal{D}^m}\left(L_{\mathcal{D}}(A(S)) \geq \frac{1}{8}\right) \geq \frac{4}{28} = \frac{1}{7}$$

□

Como consecuencia del teorema, podemos decir que no hay un algoritmo de aprendizaje óptimo para todas las distribuciones, puesto que para una dada por el resultado del teorema, el algoritmo ERM con $\mathcal{H} = \{f\}$ aprendería mejor.

Referencias

[1] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning*, ser. nil. Cambridge University Press, 2014. [Online]. Available: http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning